

**IN THE SPECIFICATION:**

**Please delete paragraph [0005] and substitute therefore the following:**

[0005] There are other known mechanisms that provide web-based instant notification. One type of such mechanisms includes online instant messaging or online chatting. Mechanisms in this category rely on proprietary protocols and deliver mechanisms, both of which can not be easily incorporated into web-based enterprise applications. A different category of mechanisms includes various remote messaging mechanisms such as Remote Procedure Call (~~PRC~~) (RPC), Common Object Request Broker (CORBA) architecture, JAVA Remote Method Invocation (RMI), and Java Messaging Service (JMS). Since the mechanisms in this category are initially designed for client/server applications, although efforts are made to utilize them in web environment, such efforts have so far proven to be difficult due to reasons such as restrictions imposed by firewalls and the highly distributed and multi-platform nature of the Internet.

**Please delete paragraph [0038] and substitute therefore the following:**

[0038] In FIG. 2, the RMF server 160 handles requests from the web clients, listens for events that are subscribed by web clients with respect to the message board objects, and multicasts the events to appropriate clients according to their subscriptions. The RMF server 160 may operate as an extension to the web server 155 as a servlet if the web server 155 supports ~~serlet~~ servlet. The RMF server 160 may also operate as a stand-alone server connected to the web server 155 through a well-defined interface. For example, a stand-alone RMF server may interact with a web server through a Common Gateway Interface (CGI).

**Please delete paragraph [0039] and substitute therefore the following:**

[0039] In FIG. 2, the RMF server 160 comprises a session manager 230, a channel manager 235, a message parser 240, one or more listener agent 245, a based ~~filer~~ filter agent 250, a producer registry 255, a message board 260, and an RMF server API 265, and an access control profile 270. The event producer 180 interfaces with the RMF server 160 through the RMF server API 265. The interface may allow the event producer 180 to register itself in the producer registry 255 and to publish data in the message board 260. Through the RMF server API 265, the RMF server 160 may also request the event

producer 180 to authenticate a web client and to filter certain events. In Appendix B, an exemplary RMF server API is incorporated as part of the present invention.

**Please delete paragraph [0048] and substitute therefore the following:**

[0048] Prior to sending an event to the connected channel, the listener agent 245 may check with a base filter agent 250 to perform certain filtering operation for access control purposes. It may also further check with a filter agent 290 within an event producer (e.g., 180) to perform dynamic event filtering. The filtered event is then sent from the listener agent 245 to the channel dedicated to the client that subscribes the event. At this point, the channel ~~manger~~ manager 235 dispatches the event to an appropriate web client.

**Please delete paragraph [0056] and substitute therefore the following:**

[0056] To receive events, an event listener may be registered with the message board 260 along with an event name. A listener may be associated with more than one event types. For example, a listener ~~gent~~ agent may be

registered to listen to an insertion data event on a slot. Events may be sent out synchronously or asynchronously, depending on the particular setting of a slot. A default mode may be set as synchronous. A slot event may be SLOT\_CREATED, SLOT\_CLEARED or SLOT\_DELETED. Slot events only specify the slot name associated with an event. A data event may be DATA\_POSTED, DATA\_CHANGED or DATA\_DELETED. This category of events (data event) may be registered with both a slot name and a data item reference.

**Please delete paragraph [0087] and substitute therefore the following:**

[0087] FIG. 9 describes exemplary schematics of a process, in which an event producer connects itself to the RMF server 160. The event producer (e.g., 180) first registers with the message board 260. This may lead to the creation of a new slot in the message board. The event producer 180 may also register a session agent with the RMF server 160. The registered session agent may be used to perform authentication on the web clients that intend to subscribe an event associated with the event producer. In addition, as part of ~~registration~~ registration, the event producer may also register a listener agent with the RMF server 160 and the listener agent may be associated with the slot that is registered under the event producer.